E801795

2

AFATL-TR-88-115

Program EAGLE Plotting System (EPS) User's Manual

AD-A204 140

Rudy A Johnson
Dave M Belk

AERODYNAMICS BRANCH
AEROMECHANICS DIVISION

SEPTEMBER 1988

FINAL REPORT FOR PERIOD SEPTEMBER 1986-SEPTEMBER 1988

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED

AIR FORCE ARMAMENT LABORATORY
Air Force Systems Command ▪ United States Air Force ▪ Eglin Air Force Base, Florida

88 10 18 013

NOTICE

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely related Government procurement operation, the United States Government thereby incurs no responsibility nor any obligation whatsoever; and the fact that the Government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication or otherwise as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

This report has been reviewed by the Public Affairs Office (PA) and is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nations.

This technical report has been reviewed and is approved for publication.

FOR THE COMMANDER

STEPHEN C. KORN
Chief, Aeromechanics Division

If your address has changed, if you wish to be removed from our mailing list, or if the addressee is no longer employed by your organization, please notify AFATL/FXA , Eglin AFB FL 32542-5434.

Copies of this report should not be returned unless return is required by security considerations, contractual obligations, or notice on a specific document.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION/AVAILABILITY OF REPORT Approved for Public Release; distribution is unlimited.
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S)	5. MONITORING ORGANIZATION REPORT NUMBER(S) AFATL-TR-88-115
6a. NAME OF PERFORMING ORGANIZATION Aerodynamics Branch Aeromechanics Division 6b. OFFICE SYMBOL (If applicable) AFATL/FXA	7a. NAME OF MONITORING ORGANIZATION Aerodynamics Branch Aeromechanics Division
6c. ADDRESS (City, State, and ZIP Code) Air Force Armament Laboratory Eglin Air Force Base, Florida 32542-5434	7b. ADDRESS (City, State, and ZIP Code) Air Force Armament Laboratory Eglin Air Force Base, Florida 32542-5434
8a. NAME OF FUNDING/SPONSORING ORGANIZATION 8b. OFFICE SYMBOL (If applicable)	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
8c. ADDRESS (City, State, and ZIP Code)	10. SOURCE OF FUNDING NUMBERS

PROGRAM ELEMENT NO.	PROJECT NO.	TASK NO	WORK UNIT ACCESSION NO.
62602F	2567	03	08

11. TITLE (Include Security Classification)

Program EAGLE Plotting System (EPS) User's Manual

12. PERSONAL AUTHOR(S)

Rudy A. Johnson, Dave M. Belk

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Year, Month, Day)	15. PAGE COUNT
Final	FROM Sep 86 TO Sep 88	September 1988	51

16. SUPPLEMENTARY NOTATION

Availability of report is specified on verso of front cover.

17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	2&3 D Graphics Contours Velocity Vectors

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

This User's Manual was designed to display the grids and flow solutions generated by the Program EAGLE Elliptic Grid Generator and Flow Solver. EAGLE Plotting System (EPS) is an interactive three-dimensional color graphics system that runs on the Tektronix 4128/4129 terminals. Many plotting operations are available, including grid plots, surface plots, six types of contours, and velocity vectors. A full description of inputs is given along with examples.

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT ☐ UNCLASSIFIED/UNLIMITED ☒ SAME AS RPT. ☐ DTIC USERS	21. ABSTRACT SECURITY CLASSIFICATION UNCLASSIFIED	
22a. NAME OF RESPONSIBLE INDIVIDUAL Rudy Johnson	22b. TELEPHONE (Include Area Code) (904) 882-3124	22c. OFFICE SYMBOL AFATL/FXA

DD Form 1473, JUN 86 — Previous editions are obsolete.

SECURITY CLASSIFICATION OF THIS PAGE

UNCLASSIFIED

PREFACE

This program was conducted in house by the Computational Fluid Dynamics Section of the Air Force Armament Laboratory, Eglin Air Force Base FL 32542-5434. Dr. Dave Belk managed the program for the Armament Laboratory. This program was conducted during the period from September 1986 to September 1988.

TABLE OF CONTENTS

Section	Title		Page
I	INTRODUCTION		1
II	USING EPS		4
	1 THE GRID FILE		4
	2 THE RESTART FILE		5
	3 THE SURFACE FILE		5
III	COMMAND REFERENCE		6
	1. *CCM	CHANGE COLOR MAP	7
	2. *CLS	CLEAR SCREEN	8
	3. *CON	DRAW CONTOURS	8
	4. *CRS	CREATE A RUN STREAM	9
	5. *DAS	DELETE A SEGMENT	10
	6. *EXIT	EXIT EPS	10
	7. *EXT	EXTRAPOLATION OF CONTOURS	10
	8. *GRD	PLOT GRID	10
	9. *HELP	HELP	11
	10. *KEY	LOAD KEYS	11
	11. *LDS	LOAD SEGMENTS	12
	12. *MSV	MAKE ALL SEGMENTS VISIBLE	12
	13. *NCM	NEW COLOR MAP	12
	14. *NSG	SET SEGMENT NUMBER	13
	15. *OUT	PRINT DATA	13
	16. *QUIT	EXIT EPS	14
	17. *RGP	REGROUP SEGMENTS	14
	18. *RSI	RUN STREAM INPUT	14
	19. *RSV	RETURN TO SINGLE VIEW	15
	20. *SAL	SET AMBIENT LIGHT	15
	21. *SBP	SET BLOCK POSITION	15
	22. *SDL	SET DIRECTED LIGHT	15
	23. *SFC	PLOT SURFACE	17
	24. *SPP	SET CONSTANT PLANE POSITION	18
	25. *SSD	SAVE SEGMENTS TO DISK	18
	26. *SSP	SET SEGMENT POSITION	19
	27. *SSV	SET SEGMENT VISIBILITY	19
	28. *STC	SET TEXT COLOR	19
	29. *STF	SET TEXT FONT	20
	30. *STS	SET TEXT SIZE	20
	31. *SVV	SCALE VELOCITY VECTORS	20
	32. *TEX	GRAPH TEXT	21
	33. *TSO	TURN SEGMENT ON	21
	34. *4VW	4 - VIEW	21
IV	USING RUN STREAMS		23
	REFERENCES		43

TABLE OF CONTENTS (Concluded)

Section	Title	Page
Appendix		
A	PROGRAM TO GENERATE EXAMPLE DATA FILES	27
B	EPS FORTRAN SOURCE CODE	31

LIST OF FIGURES

Figure	Title	Page
1	Terminal AXIS orientation	16
2	NACA 0012 density contours.	26

LIST OF TABLES

Table	Title	Page
1	Program EAGLE - Plotting system commands	6
2	Surface command summary	16

SECTION I
INTRODUCTION

Program EAGLE - Plotting System (EPS) is an interactive two- and three-dimensional plotting system designed to display the computational grids produced by Program EAGLE - Numerical Grid Generation System and the flow solutions produced by the Program EAGLE - Flow Solver. (See References 1 and 2.)

This program was designed to be used with the Tektronix 4128/4129 graphics terminals while running on a VAX/VMS or CRAY-2/CTSS host computer. EPS allows the operator to plot grids or surfaces and to draw contours or velocity vectors on any constant curvilinear coordinate plane or portion of a plane.

The Tektronix 4128/4129 supports definition of graphics objects in the terminal memory. These objects, called segments, can then be manipulated locally by the terminals. Each object EPS draws to the screen is assigned a unique segment number. This is done for every constant plane of grid, contours, or velocity vectors and for any text that is inserted into the graphics area. Both terminals have local viewing functions that are controlled by the thumb wheels. In the three-dimensional mode some of the options included are zoom, pan, and rotate operations. These local viewing functions do not have any affect on the two-dimensional segments that EPS defines for the text and the scale of the contour and the velocity vector plots. Another terminal capability used by EPS is the Graphics input (Gin). The two Gin functions used are the pick and the locate operations. The pick operation allows the user to choose a visible segment with the thumb wheels for a specified operation. The locate operation allows the user to move a segment that already exists. The two Tektronix terminals mentioned above are very similar; however, the notable difference concerning EPS is that the 4128 terminals do not support solid surface shading. This also means that the hidden line option is not available on the 4128 terminals.

The grid plotting portion of EPS allows the operator to plot the surface and the grid files by specifying a constant plane or a line contained in the file. The grid plotter will also plot the grid portion of the restart file. The inputs used in plotting the grid and the restart files are the same. When plotting the surface file, a different set of inputs are required because these files only contain the two-dimensional surfaces.

The contouring section of EPS plots the flow solutions that are stored in the flow solver restart files. The six types of contours that can be plotted are Mach, density, pressure, dynamic pressure, pressure coefficient, and energy. The inputs to EPS for the contour plotting are very similar to those used when plotting grids. When drawing contours, a gap occurs at the block boundaries because the grid image points are not stored in the restart file. EPS has an option that will hide these block boundaries by making a second order extrapolation of the grid points at these boundaries. The results of this approximation will vary depending on the slope of the contour lines as they approach the boundaries.

The velocity vectors are also plotted from the restart file and the inputs are the same as those for drawing the contours. EPS draws three dimensional velocity vectors that are scaled and colored according to the magnitude of the total velocity at each cell. Since the EAGLE - Flow Solver makes its calculations at the cell centers, the starting point of each velocity vector is positioned at the corresponding cell center. Once the velocity vectors are drawn, their length can be scaled by the operator. This scaling changes the size of every velocity vector by the same amount maintaining their relative sizes.

EPS also supports run stream input. The run stream file contains all inputs that an operator would make while running interactively. To use run stream input the user simply activates EPS then selects the run stream input option. When the end of the run stream is detected, EPS returns control to the operator. An option also exists that allows EPS to create a journal file that contains all inputs that are made interactively or by a run stream.

EPS also contains several options to enhance the display. One of the most useful is the introduction of text into the graphics area. This text can appear in multiple sizes, colors, and fonts, and as previously mentioned it is not effected by the the terminals local viewing functions. Another available option is modifying or completely replacing the color map used in drawing contours and velocity vectors. When using the 4129 terminals, EPS can also define up to 16 light sources to highlight specific areas of the shaded surfaces. EPS also gives the operator the ability to position existing segments with the thumb wheels. These locating operations can be performed on individual segments or on groups of segments.

In the following chapters the operation of EPS will be discussed by first describing how the data is arranged in the files, then by giving a description of the commands with short examples for those commands that require it. Following this will be a description of run stream usage.

Appendix A contains a FORTRAN program that will generate the three types of data files and Appendix B contains a description of the EPS subroutines.

SECTION II
USING EPS

When EPS is executed, it starts by initializing the terminal for three-dimensional graphics and prompting the user for a viewing volume radius. This radius is the scaling factor for all coordinates and its value should be approximately that of the largest point that will be plotted. If the radius exceeds the largest point by significant amount resolution, problems will occur. When graphics already exist, the same viewing radius should be used that was in effect when the existing graphics was plotted.

Once this is completed the standard prompt will then appear.

```
................
COMMAND :
```

From this point any command can be issued and EPS will prompt for the various inputs that are unique to each command. For example if unwanted graphics exist a "*CLS" would be entered to remove it.

The next three paragraphs will describe the data files that EPS is designed to plot.

1. The Grid File

This is the format of the file which is created by the EAGLE Grid Code. The file is unformatted with the first line containing the number of blocks and a set of the three maximum indices for each block. The remainder of the file contains the three coordinate values of a grid point on each record.

The grid file is arranged as follows, with each line representing a separate FORTRAN record.

```
bmax imax jmax kmax ... imax(bmax) jmax(bmax) kmax(bmax)
x(1,1,1) y(1,1,1) z(1,1,1)
x(2,1,1) y(2,1,1) z(2,1,1)
   .        .        .
   .        .        .
   .        .        .
```

When attempting to create this file, the sample program in Appendix A will be helpful.

2. The Restart File

This data file is created by the EAGLE Flow Solver. The first line is the same as that of the grid file. Line two contains the number of iterations the flow solver has made to produce the data in the file. This quantity is not used by EPS, but it must be present in the data file. The third line contains the six "LAP" values which are used in dimensioning the six flow field variables. This line will occur before the data for each block. The next nine lines contain the grid points (first three lines) and the six flow field variables. These variables are density, density times the x-velocity component, density times the y-velocity component, density times the z-velocity component, energy, and pressure.

The restart file is arranged as follows.

```
bmax imax jmax kmax ... imax(bmax) jmax(bmax) kmax(bmax)
istep
0 1 2 3 4 5
x(1,1,1) x(2,1,1)          ...          x(imax,jmax,kmax)
y(1,1,1) y(2,1,1)         ...          y(imax,jmax,kmax)
z(1,1,1) z(2,1,1)          ...          z(imax,jmax,kmax)
density ...
density times X-velocity ...
density times Y-velocity ...
density times Z-velocity ...
energy ...
pressure ...
```

See Appendix A to generate an example restart file.

3. The Surface File

The format of the files generated by the surface portion of the EAGLE - Grid Code are similar to the grid files. The only difference is that the surface files do not contain the dimensions on the first line.

SECTION III

COMMAND REFERENCE

TABLE 1. PROGRAM EAGLE - PLOTTING SYSTEM COMMANDS

COMMAND	OPERATION	PAGE
+ *CCM	CHANGE COLOR MAP	8
+ *CLS	CLEAR SCREEN	9
*CON	DRAW CONTOURS	9
*CRS	CREATE A RUN STREAM	10
+ *DAS	DELETE A SEGMENT	11
+ *EXIT	EXIT EPS	11
*EXT	EXTRAPOLATION OF CONTOURS	11
*GRD	PLOT GRID	11
+ *HELP	HELP	12
*KEY	LOAD KEYS	12
*LDS	LOAD SEGMENTS	13
*MSV	MAKE ALL SEGMENTS VISIBLE	13
*NCM	NEW COLOR MAP	13
*NSG	SET SEGMENT NUMBER	14
*OUT	PRINT DATA	14
+ *QUIT	EXIT EPS	15
*RGP	REGROUP SEGMENTS	15
*RSI	RUN STREAM INPUT	15
*RSV	RETURN TO SINGLE VIEW	16
*SAL	SET AMBIENT LIGHT	16
*SBP	SET BLOCK POSITION	16
*SDL	SET DIRECTED LIGHT	16
*SFC	PLOT SURFACE	17
*SPP	SET CONSTANT PLANE POSITION	19
+ *SSD	SAVE SEGMENTS TO DISK	19
*SSP	SET SEGMENT POSITION	19
+ *SSV	SET SEGMENT VISIBILITY	20
*STC	SET TEXT COLOR	20
*STF	SET TEXT FONT	20
*STS	SET TEXT SIZE	21
*SVV	SCALE VELOCITY VECTORS	21
*TEX	GRAPH TEXT	21
*TSO	TURN SEGMENT ON	22
*4VW	4 - VIEW	22

+ These commands are not valid in a run stream.

1. CCM Change the Color Map

This command allows the user to manually change the colors of the contours, velocity vectors, and the shaded surfaces. When issued, EPS prompts for either a "C" to change the contour or velocity vector color map or an "S" to change the shaded surface color map. The range of indices are displayed along with the hue, lightness, and saturation (HLS) values that are currently active. The hue can range from 0 to 360 degrees. The lightness and saturation range from 0 to 100, where 100 is the lightest or most saturated. (See Reference 3 or 4.) The user selects the range of indices that are to be changed then enters the HLS values for the first and last index selected and a linear distribution is used to assign values to the HLS coordinates between the selected range. The new color map takes effect immediately and the user enters a "C" to continue making changes or a "Q" to quit making changes and save the new color map to a data file.

EXAMPLE

```
..........
COMMAND :
*ccm
CHANGE (C) CONTOUR OR (S) SHADED SURFACE COLOR MAP
c
CURRENT HLS VALUES ARE
132   120   50   75
133   121   50   75
 .     .     .    .
 .     .     .    .
255   360   50   75
ENTER THE RANGE OF INDICES TO BE CHANGED
140 220
ENTER THE HUE, LIGHTNESS, AND SATURATION FOR  140
140 50 60
ENTER THE HUE, LIGHTNESS, AND SATURATION FOR  220
380 70 90
ENTER (Q) TO QUIT OR (C) TO CONTINUE
q
ENTER THE NEW COLOR MAP FILE NAME
color.map
..........
COMMAND :
*ccm
```

```
CHANGE (C) CONTOUR OR (S) SHADED SURFACE COLOR MAP
s
ENTER THE GRID COLOR
2
CURRENT HLS VALUES ARE
32   40   50   75
.    .    .    .
.    .    .    .
.    .    .    .
```

2. *CLS Clear Screen

When issued this command causes all segments to be deleted, the current files to be closed, and the segment number is set to four.

3. *CON DRAW CONTOURS

If this is the first plotting command issued, then the restart file name is requested. The first time that the CON command is given EPS prompts for the type of contour or velocity vectors to be drawn and a pause will occur while EPS cycles through the restart file searching for the minimum and maximum values. These values are then displayed and the user may change them. The next entry is the spacing between contours. When velocity vectors have been selected, this entry is not present. If this number is too small, the terminals memory will be exceeded because of the large number of contours drawn. To increase the graphics capabilities of the terminal set the dialog area buffer size to 50. If the restart file contains more than one block, EPS prompts for the block to be plotted from. The user then specifies the upper and lower limits of the plane that is to be contoured. Note that the constant plane should never be one because EPS would try to draw contours onto the grid image points which are not physically meaningful and the resulting plot would be useless. Thus, EPS will not contour a constant one plane but will issue a warning.

EXAMPLE

```
..........
COMMAND :
*con
ENTER THE FILENAME
test.dat
SELECT THE TYPE OF CONTOUR PLOT
  1 - DYNAMIC PRESSURE
  2 - MACH
  3 - PRESSURE
  4 - DENSITY
  5 - ENERGY
  6 - PRESSURE COEFFICIENT
  7 - VELOCITY VECTORS
2
CALCULATING MIN AND MAX
MINIMUM MACH = 0.267
MAXIMUM MACH = 1.697
ARE THESE VALUES GOOD (Y/N)
y
ENTER THE SPACING BETWEEN CONTOURS
THE DEFAULT IS  0.00196
.002
BLOCK IMAX  JMAX  KMAX
  1    25    10    19
  2    30    17    22
ENTER THE BLOCK TO BE PLOTTED FROM
2
* * * * * * * * * * * * * * * * * * * *
 BLOCK NUMBER:  2
   IMAX = 30   JMAX = 17   KMAX = 22
ENTER THE PLANE TO BE PLOTTED
NOTE ONE OF THE DIMENSIONS MUST REMAIN CONSTANT
1 1 2 30 17 2
NO EXTRAPOLATION OF CONTOURS
..........
```

4. *CRS   Create Run Stream

When issued, this command causes EPS to prompt for a file name which is used to save information entered through the keyboard or from a run stream. This command can be entered at any time, but should only be used once. Only commands that are valid in a run stream (see Table 2) and are entered after the CRS command will be saved. It is important to note that the viewing volume radius is not saved in the run stream.

5. *DAS Delete A Segment

When issued, EPS prompts the user to position the cursor on the segment and press the space bar. The segment begins to flash and the user is asked if the correct segment is flashing. If the answer is yes the segment is deleted, else it is returned to its previous status. Once a segment is deleted the only way to get it back is to replot it.

6. *EXIT Exit

Issue this command to terminate the execution of EPS. The *QUIT command is also available.

7. *EXT Extrapolation of Contours

This command allows EPS to hide the gaps between the contours which are caused by the block boundaries. This command uses a second order extrapolation of the cell centered coordinates at the block boundaries and only contours plotted after the command has been issued are affected. Since this is a simple extrapolation, the results will vary depending on the slope of the contour lines as they approach the block boundaries. The default is no extrapolation.

8. *GRD Plot Grid

This command is used to plot a constant plane from a specified block of a grid or restart file. The first time this command is given EPS will prompt for the file name and file type. When the file to be plotted contains more than one block, EPS prompts for the block to be plotted from. Then the lower and upper limits of the plane to be plotted are entered along with a seventh number that is used to determine the direction of the outward normals to the plane. This number only affects the shaded surfaces and should be zero for a constant plane with the index increasing in the direction of the outward

normal. When the index decreases in the outward normal direction, the normal specifier should be one. EPS then prompts for the color of the plane. If at any time no plane is to be drawn enter a "/".

EXAMPLE

```
..........
COMMAND :
*grd
ENTER THE FILENAME
test.dat
ENTER THE FILE TYPE
  1   GRID FILE
  2   RESTART FILE
1
BLOCK IMAX  JMAX  KMAX
  1    25    10    19
  2    30    17    22
ENTER THE BLOCK TO BE PLOTTED FROM
2
* * * * * * * * * * * * * * * * * * * * 
 BLOCK NUMBER:  2
   IMAX = 30   JMAX = 17   KMAX = 22
ENTER THE PLANE TO BE PLOTTED
NOTE ONE OF THE DIMENSIONS MUST REMAIN CONSTANT
1 1 1 30 17 1 0
COLOR TABLE
1...WHITE
2...RED
3...GREEN
4...BLUE
5...CYAN
6...MAGENTA
7...YELLOW
ENTER THE COLOR OF THIS PLANE
2
..........
```

9. *HELP

HELP causes a list of commands to be printed on the screen.

10. *KEY Program Function Keys

This causes EPS to program the function keys with the following display options.

F1 - Two by two dithering.
S1 - Halftoning with two by two dithering.
F2 - Four by four dithering.
S2 - Halftoning with four by four dithering.
F3 - Outline on.
S3 - Outline off.
F4 - Hidden line on.

11. *LDS Load Segments

This command causes segments that are saved to the terminal's disk with the SSD command to be drawn on the screen. Before using this command the CLS command should be used to insure that no segments exist which have the same segment number as one that is stored on the disk. The local hard drives are labeled S0: and S1: and the local floppy drives are F0: and F1:.

EXAMPLE

```
..........
COMMAND :
*lds
ENTER THE LOCAL DISK FILE NAME
s0:test.12n
```

12. *MSV Make all Segments Visible

This command makes all existing graphic segments visible except the cursor and the fundamental velocity vector which are defined when EPS is executed.

13. *NCM New Color Map

This command is used to replace the default color map that is contained in EPS with a color map stored on a host computer file. The color map file format should be the same as that created by EPS.

The file format is as follows:

```
inf inl
index hue light sat
  .    .    .    .
  .    .    .    .
  .    .    .    .
```

Where inf is the first index value, inl is the last index value, index ranges between inf and inl and each value is associated with a particular color, and the three remaining variables are the hue, lightness, and saturation values that are associated with the color coordinate cone that is described in References 3, 5, and 6.

14. *NSG Set Segment Number

This command is used to specify the segment number which is assigned to the next graphics object which appears on the screen. EPS uses segment numbers one through three and will not accept these values. When EPS is executed, the segment number is set to four. It is also set to four when the *CLS command is given.

When segments already exist, it is important to specify the segment number in order to avoid assigning the same segment number to two segments. If this should occur, the terminal will detect an error and the new segment will be drawn without a segment number.

15. *OUT Print Data

This command is used to extract and print to a file a specified range of coordinates from the grid, surface, or restart files. In addition velocities or one of the quantities that can be contoured may be printed from the restart file. When issued, EPS will prompt for the quantity to be printed. The user then specifies printing to the terminal, to a file, or to both. When a file is used the user is prompted for a name.

EXAMPLE

```
..........
COMMAND :
*out
OUTPUT  1 - GRID POINTS
        2 - VELOCITY
        3 - CONTOUR
        4 - OFF
1
OUTPUT TO  1 - SCREEN
           2 - FILE
           3 - BOTH
2
ENTER THE FILENAME
dat.out
```

The next step is to use the grid plotting commands to cause the coordinates to be printed or the *CON command to cause the selected quantity to be printed. The range that is printed is given when EPS prompts for the dimensions of the constant plane that is to be plotted.

16. *QUIT Exit EPS

Issue this command to terminate the execution of EPS. *EXIT can also be used.

17. *RGP Regroup Segments

This command returns all segments to the position that they occupied previous to the execution of the SBP, SPP, or SSP commands.

18. *RSI Run Stream Input

This command causes EPS to prompt for the run stream name. EPS reads the file the same as it would if the information were being entered through the keyboard. For help in creating a run stream see the CRS command and the using run stream section.

19. *RSV Return to Single View

This command returns the view which was present before the 4VW command was issued.

20. *SAL Set Ambient Light

This command is used to set the ambient light. The effects of this command can only be seen when the shaded surface is on and the change is not made until the view is refreshed. The range is zero to one, where one is the brightest. The ambient light is set to .3 when the EPS is executed.

21. *SBP Set Block Position

This command causes EPS to prompt the user to position the cursor where the plots from block one are to be located and press the space bar. Then the plots from the next block are positioned. This process continues until all blocks have been positioned. To position the segments as they were previous to this command use the RGP command.

22. *SDL Set Directed Light

This command is used to set the intensity and location of the light sources. Two light sources are automatically turned on when the EPS is executed. (source =1, uvw = -10, 10, 5, intensity = .7, source = 2, uvw = 10, 10, 5, intensity = .3) Sixteen light sources are available to be defined by the user. The intensity of these sources can range from zero to one and the location can be in either uvw terminal coordinates or in xyz coordinates. The uvw coordinates rotate with the plot while the xyz coordinates remain fixed. See Figure 1 for the orientation of these axes.

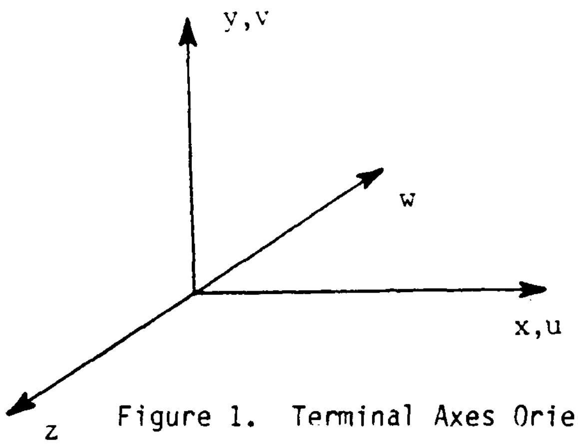


Figure 1. Terminal Axes Orientation

EXAMPLE

```
..........
COMMAND :
*sdl
ENTER SOURCE NUMBER
1
XYZ OR UVW COORDINATES
x
ENTER SOURCE POSITION
DEFAULT IS -10 10 5
-10 25 15
ENTER INTENSITY
.8
..........
```

TABLE 2. SURFACE COMMAND SUMMARY

SURFACE COMMAND	OPERATION	PAGE
*END	END SURFACE PLOTTING	17
*F	FILE	18
*H	HELP	18
*I	INPUT	18
*P	PLOT	18
*R	REWIND	18

23. *SFC Plot Surface

This command is used to plot the files generated by the surface code. When *SFC is issued EPS goes into the surface mode and a list of commands are listed. Once in this mode only surface commands, listed in Table 2 are valid and the prompt will change to the following.

....
SFC COMMAND :

While in the surface mode, the six commands that follow are permitted.

(a) *END End Surface Plotting

This command returns control to the main program.

(b) *F File

This command is used to specify the surface file that is to be plotted. This command can be used several times to connect different files.

(c) *H Help

This command gives a list of surface commands.

(d) *I Input

This is how the range of values that are read from the surface file are specified. The two dimensions of the surface are entered when this command is issued.

(e) *P Plot

When issued this command causes EPS to prompt for the color of the surface to be plotted then the plane that was specified by the latest *Input command will be plotted.

(f) *R Rewind

This command rewinds the connected surface file.

EXAMPLE

```
..........
COMMAND :
*sfc

..............
SCF COMMAND :
*f
ENTER THE FILENAME
sfc.dat
..............
SCF COMMAND :
*i
ENTER THE DIMENSIONS, THE INNER LOOP IS FIRST
30 10
..............
SCF COMMAND :
*p
COLOR TABLE
1...WHITE
2...RED
3...GREEN
4...BLUE
5...CYAN
6...MAGENTA
7...YELLOW
ENTER THE COLOR OF THIS PLANE
2
..............
SCF COMMAND :
*end
..........
COMMAND :
```

24. *SPP Set Constant Plane Position

This command is similar to the SBP command, with the difference being the constant I, J, and K plane positions are set. To position the segments as they were previous to this command use the RGP command.

25. *SSD Save Segments to Disk

This command is used to save all currently defined segments to one of the local disk drives. For information on the disk names and loading the segments to the screen see the LDS command.

EXAMPLE

```
..........
COMMAND :
*ssd
ENTER THE LOCAL DISK FILE NAME
s0:test.12n
```

26. *SSP Set Segment Position

SSP causes EPS to prompt for the cursor to be placed on a visible segment and to then press the space bar. This starts the picked segment flashing and a check is made to be sure that the correct segment has been selected. If the picked segment is graph text or the contour scale, then it is moved with the thumb wheels; however, if the picked segment is a constant plane, then the cursor is positioned and when the spacebar is pressed the segment appears at the new location.

27. *SSV Set Segment Visibility

When issued, EPS prompts the user to position the cursor on the graphics segment and press the space bar. The segment will begin to flash and the user is prompted to turn the segment on or off. This command does not delete the segment. It simply turns it off. Once the segment is turned off the only way to make it visible again is to use the TSO command or the MSV command.

EXAMPLE

```
..........
COMMAND :
*ssv
POSITION CURSOR WITH THUMB WHEELS
THEN PRESS THE SPACE BAR
SEGMENT NUMBER = 3
ENTER THE VISIBILITY: 0 - OFF
                      1 - ON
1
..........
```

28. *STC Set Text Color

This command allows the user to set the color of all text entered after this command.

29. *STF Set Text Font

This command allows the user to chose from any of the fonts that are available on the terminal. Once this command is issued all text will be printed in the selected font until a new font is selected. Note that at times a Tektronix graphics error will be caused if the font to be loaded from the hard disk has been previously loaded to the terminal. The text to be placed on the screen should not be affected by the error.

Since EPS does not contain any font files the fonts available will depend on those that are on the particular terminal. EPS uses the following character string to access the fonts.

SO:F##.FNT

Where the ## are the specific font selected. If the fonts on a particular machine do not use this naming scheme then this string will need to be changed. (See subroutine TEXT in Appendix B.) If the font selected is not found the default keyboard font will be used.

30. *STS Set Text Size

This command allows the user to set the text height, width and spacing between characters. This command remains in effect until the next time it is issued.

31. *SVV Scale Velocity Vectors

This command causes EPS to scale all velocity vectors by a scaling factor entered by the user. The scaling done by this command is not additive. For example if the velocity vectors are to be doubled in size a "2" is entered. Now to get the vectors back to their original size enter a "1". It is also important to note that no change is apparent until the screen is refreshed.

32. *TEX Graph Text

This command is used to label plots. EPS prompts for the graph text and has the user position it on the screen. See the STC command to set the text color, the STS command to set the text size, and the STF command to set the text font. The text is positioned with the thumb wheels. It should also be noted that the text is not affected by the terminals built in graphics manipulation options. Note that at times a Tektronix graphics error will be caused if the font (selected by the STF command) to be loaded from the hard disk has been previously loaded to the terminal. The text to be placed on the screen should not be affected by the error.

33. *TSO Turn Segment On

This command will cause EPS to prompt for the segment number to be turned on and then the segment will appear. If the segment was turned off with the SSV command its number was given.

34. *4VW 4 - View

When issued the operator is directed to position the cursor on any of the segments that are to be visible in the first window and press the space bar. To include all segments press an "a" and to move to the next window press a "/".

EXAMPLE

```
..........
COMMAND :
*4vw
SELECT SEGMENTS TO BE DISPLAYED IN QUADRANT  1
PRESS " " TO INCLUDE A SEGMENT.
PRESS "A" TO INCLUDE ALL SEGMENTS.
PRESS "/" TO MOVE TO THE NEXT QUADRANT.
POSITION CURSOR WITH THUMB WHEELS
< / >
SELECT SEGMENTS TO BE DISPLAYED IN QUADRANT  2
```

The windows have been labeled as the four quadrants of a two dimensional Cartesian coordinate system. When the space, A, or / are pressed they do not appear on the screen and a slight delay occurs while EPS processes information. When the selections are finished for the fourth window the screen is divided and the selected segments become visible. Now to move from one window to the next use the next view key. All keyboard viewing functions will work in the current window and will not effect the segments in the other quadrants.

SECTION IV

USING RUN STREAMS

There are two ways to create a run stream. The first is to use the CRS command in EPS. When this method is used all commands that are valid in a run stream are written to a file as they are issued. The second method is to create the run stream in an editor. When creating the file, all commands and alphabetic characters should start on the first line. Numeric characters should start on the second column or further to the right.

The following is a line by line description of each command available for use in a run stream file.

```
*CON
RESTART.DAT ( omit the filename if this is not the first time
              one of the restart plotting routines is used
              GRD or CON )
 3          ( contour type three is selected omit if this is
              not the first time CON is used )
N           ( the range of contours are not satisfactory omit
              if this is not the first time CON is used )
 0.03 1.03  ( this line would not be present if the above
              answer were "Y", these are the new minimum and
              maximum contour values omit if this is not the
              first time that CON is used )
 0.02       ( spacing between contours omit if this is not the
              first time CON is used )
 1              ( the block to be plotted from, omit for the
                  one block case )

 1 1 2 10 18 2  ( the three starting and ending points of
                  the plane to be contoured )

*CRS
 runst.dat      ( run stream name to be created )

*EXT
 1              ( 0 - no extrapolation, 1 - second order )

*GRD
TEST.DAT          ( omit the filename if this is not the first
                    time the GRD or the CON command is used )
 2                ( file type 1 - grid, 2 -  restart, omit if
                    this is not the first plotting command used)
 1                ( the block to be plotted from, omit for the
                    one block case )
```

```
 1 1 1 10 18 1 0   ( the three starting and ending points of
                     the plane to be plotted and the number to
                     determine the direction of the normal )
 2                 ( the color of the plane to be plotted )

*KEY               ( No inputs. )

*LDS
SO:PLOT.SEG      ( local disk drive and filename to that
                   contains the graphics segments )
*MSV             ( No inputs. )

*NCM
COLOR.MAP        ( name of the color map file )

*NSG
 10              ( next assigned segment number )

*OUT
 2               ( quantity that is to be printed )
 3               ( location the output is to be sent to )
out.dat          ( output filename, not present if output
                   is directed to the screen )

*RGP             ( No inputs. )

*RSI
 RUNST.DAT       ( name of run stream )

*RSV             ( No inputs. )

*SAL
 0.5             ( ambient light intensity )

*SBP
 100 220 50        ( x, y, z  position of blocks )
 600 100 100       ( there should be one set of coordinates
  .   .   .          for each block of grid )
  .   .   .
  .   .   .

*SDL
 1              ( source number )
x               ( xyz or uvw coordinates )
 10 10 -3       ( position in the selected coordinate system )
 0.7            ( intensity of the source )

*SFC            ( No inputs. )

*F              ( only valid once the SFC command is issued )
surf.dat        ( surface file name )
```

```
*I               ( only valid once the SFC command is issued )
 10 20           ( dimensions of the surface )

*P               ( only valid once the SFC command is issued )
 2               ( surface color )

*R               ( no inputs, only valid once the SFC command is
                   issued )

*SPP
 102 123 145     ( x, y, z positions of the constant planes )
 659 220 100
 50  40  20

*SSP
 10              ( segment to be positioned )
 220 1500        ( new x, y position of the segment if the segment
                   is text or the contour or velocity vector scale )
 265 789 234     ( x, y, z position of the segment )

*STC
 2               ( color of following graph text )

*STF
 20              ( font number of the following text )

*STS
 120 80 30       ( the height, width, and spacing of the text )

*SVV
 2               ( velocity vector scaling factor )

*TEX
LABEL           ( string to be transformed to graph text )
 1000 1600      ( x,y location of the graph text,
                  ranges are 0 to 4095 )

*TSO
 4              ( number of the segment to be made visible )

*4VW
 2 4 5          ( the first number of each row is the number
 3 5 2 7          of segment numbers in that row, each row
 1 9              list the segments to be visible in a quadrant
 5 4 5 2 9 7      of the screen )
```

The commands that are not supported in run stream are indicated in Table 1. It is important to note that the viewing volume radius is not included in the run stream. The following is an example of a run stream to do density contours around a two dimensional airfoil (see Figure 2).

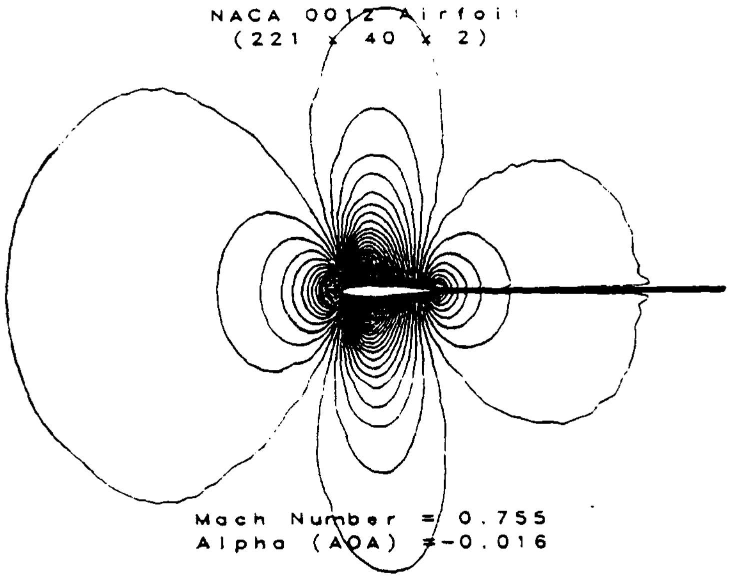


Figure 2. NACA 0012 Density Contours

```
*CON
RES.DAT
 4
Y
 .01
 1 1 2 221 40 2
*TEX
NACA 0012 Airfoil
 1348 2908
*TEX
(221 X 40 X 2)
 1348 2870
*TEX
Mach Number = 0.755
 1348 366
*TEX
Alpha (AOA) =-0.016
 1348 342
```

APPENDIX A

PROGRAM TO GENERATE EXAMPLE DATA FILES

PROGRAM TO GENERATE EXAMPLE DATA FILES

The following program will generate the three types of data files that EPS can read. The grid and restart files will contain a four block ogive cylinder body. The restart file will also contain the additional data that simulates the flow field quantities. The surface file will contain a simple two dimensional plane.

```
      PROGRAM TEST
      DIMENSION X(40,40,40),Y(40,40,40),Z(40,40,40)
      DIMENSION R(40,40,40),RU(40,40,40),RV(40,40,40)
      DIMENSION RW(40,40,40),E(40,40,40),P(40,40,40)
      DIMENSION LAP(3,2)
      INTEGER BMAX,IMAX(5),JMAX(5),KMAX(5),IB
c.. open the example restart file
      open(2,file='restart.dat',FORM='UNFORMATTED',STATUS='UNKNOWN')
c.. open the example grid file
      open(3,file='grid.dat',FORM='UNFORMATTED',STATUS='UNKNOWN')
c.. open the example surface file
      open(4,file='surface.dat',FORM='UNFORMATTED',STATUS='UNKNOWN')
c.. set the number of blocks of grid to 4
      BMAX=4
c.. in the restart file ISTEP is  the number of iterations the
c..flow solver makes
      ISTEP = 2800
c.. set the dimensions for each block
      DO 2 I=1,BMAX
      IMAX(I)=30
      JMAX(I)=2
      KMAX(I)=30
 2    CONTINUE
c.. initialize lap
      LAP(1,1)=0
      LAP(2,1)=0
      LAP(3,1)=0
      LAP(1,2)=-1
      LAP(2,2)=-1
      LAP(3,2)=-1
c.. write the number of blocks and the dimensions of each
c.. block to the restart file
      WRITE(2)BMAX,(IMAX(I),JMAX(I),KMAX(I),
     &  I=1,BMAX)
c.. write the number of blocks and the dimensions of each
c..block to the grid file
      WRITE(3)BMAX,(IMAX(I),JMAX(I),KMAX(I),
     &  I=1,BMAX)
c.. write the number of iterations to the restart file
      WRITE(2)ISTEP
```

```
c.. begin main loop
      PI=-ACOS(-1.)
      DO 50 IB=1,BMAX
      pi=-pi
c.. write the lap quantities to the restart file
        WRITE(2)((LAP(IL,JL),IL=1,3),JL=1,2)
        DO 20 I=1,IMAX(IB)
        DO 20 J=1,JMAX(IB)
        DO 20 K=1,KMAX(IB)
          IF(IB.EQ.4 .OR. IB.EQ.3)THEN
            Z(I,J,K)=FLOAT(K)/(flOAT(KMAX(IB))/2.)
            TH=PI*FLOAT(I-1)/FLOAT(IMAX(IB)-1)
            X(I,J,K)=COS(TH)
            Y(I,J,K)=SIN(TH)
          ELSE
            PH=PI*FLOAT(K-1)/(2.*FLOAT(KMAX(IB)-1))+PI/2.
            if(ib.eq.2)ph=-ph
            Z(I,J,K)=COS(PH)
            TH=PI*FLOAT(I-1)/FLOAT(IMAX(IB)-1)
            X(I,J,K)=SIN(PH)*COS(TH)
            Y(I,J,K)=SIN(PH)*SIN(TH)
          ENDIF
c.. write to the grid file
          write(3)x(i,j,k),y(i,j,k),z(i,j,k)
c.. the following are calculated to act as the six
c.. flow field properties
          R(I,J,K)=.382*J/K+1.
          RU(I,J,K)=SQRT(R(I,J,K)+x(i,j,k)*x(i,j,k))
          RV(I,J,K)=SQRT(R(I,J,K)+RU(i,j,k))
          RW(I,J,K)=SQRT(R(I,J,K)+RV(i,j,k))
          E(I,J,K)=5.*R(I,J,K)*(RU(I,J,K)+RW(I,J,K))
          P(I,J,K)=R(I,J,K)*(RU(I,J,K)+RW(I,J,K)+RV(I,J,K))
 20     CONTINUE
c.. write the grid and flow properties to the restart file
        WRITE(2)(((X(I,J,K),I=1,IMAX(IB)),J=1,JMAX(IB))
     &   ,K=1,KMAX(IB))
        WRITE(2)(((Y(I,J,K),I=1,IMAX(IB)),J=1,JMAX(IB))
     &   ,K=1,KMAX(IB))
        WRITE(2)(((Z(I,J,K),I=1,IMAX(IB)),J=1,JMAX(IB))
     &   ,K=1,KMAX(IB))
        WRITE(2)(((R(I,J,K),I=1,IMAX(IB)),J=1,JMAX(IB))
     &   ,K=1,KMAX(IB))
        WRITE(2)(((RU(I,J,K),I=1,IMAX(IB)),J=1,JMAX(IB))
     &   ,K=1,KMAX(IB))
        WRITE(2)(((RV(I,J,K),I=1,IMAX(IB)),J=1,JMAX(IB))
     &   ,K=1,KMAX(IB))
        WRITE(2)(((RW(I,J,K),I=1,IMAX(IB)),J=1,JMAX(IB))
     &   ,K=1,KMAX(IB))
        WRITE(2)(((E(I,J,K),I=1,IMAX(IB)),J=1,JMAX(IB))
     &   ,K=1,KMAX(IB))
        WRITE(2)(((P(I,J,K),I=1,IMAX(IB)),J=1,JMAX(IB))
     &   ,K=1,KMAX(IB))
```

```
 50    CONTINUE
C.. generate a surface file
        ZS=0
       DO 60 I=1,20
       DO 60 J=1,10
         XS=FLOAT(I)
         YS=FLOAT(J)
c.. write to the surface file
            write(4)XS,YS,ZS
 60    CONTINUE
       CLOSE(2)
       CLOSE(3)
       CLOSE(4)
       PRINT*,'grid, restart, and surface files have been created'
       STOP
       END
```

APPENDIX B

EPS FORTRAN SOURCE CODE

THE MAIN PROGRAM

The parameter statement that occurs on line two of the source code will need to be changed from time to time. MEMAX is the length of the ribbon vector used to store the grid points and the flow field quantities. EPS will specify the value MEMAX should be set to and stop executing when necessary. IBMAX is the number of blocks of grid EPS is able to handle and if it is exceeded execution is terminated.

```
      PROGRAM EPS
      PARAMETER ( MEMAX=500000,IBMAX=50   )
```

COMMON BLOCK DESCRIPTIONS

1. COMMON/DMNS/I1X,I1Y,I1Z,I1A,I1B,I1C,ID1,ID2,ID3,ID4,IUSED

VARIABLE	TYPE	PURPOSE
I1X,I1Y,I1Z	integer	indicate storage locations of x, y,& z points in the ribbon vector
I1A,I1B,I1C	integer	storage locations of dummy arrays used by the plotting routines
ID1,ID2,ID3, ID4	integer	storage locations of the flow field quantities
IUSED	integer	total storage used by the ribbon vector

2. COMMON/DSTN/N4,N5;N6,N7,N8,N9

VARIABLE	TYPE	PURPOSE
N4,N6,N8	integer	these are the starting points used for dimensioning the flow field variables
N5,N7,N9	integer	ending points

3. COMMON/GINGER/LSIG,LTSIG,LGFC,LGRF

VARIABLE	TYPE	PURPOSE
LSIG	integer	signature character sent by the terminal prior to a terminal report
LTSIG	integer	terminating signature character
LGFC	integer	graphics input function code
LGRF	integer	graphics input report format

4. COMMON/GLOBAL/ICORD,IABSXY

VARIABLE	TYPE	PURPOSE
ICORD	integer	current terminal coordinate mode
IABSXY	integer	specifies relative or absolute terminal coordinates

5. COMMON/PRINT/IOTPT,ISLU2

VARIABLE	TYPE	PURPOSE
IOTPT	integer	specifies which quantity is printed by *OUT
ISLU2	integer	indicates where output is sent

6. COMMON/RANGE/ FRANGE(300),NRANGE,ICOORD(4,256)

VARIABLE	TYPE	PURPOSE
FRANGE(300)	real	contains the values of the contour lines
NRANGE	integer	number of colors used when drawing contours
ICOORD(4,256)	integer	current terminal color map

7. COMMON/RXNS/NCONTOR,FSP,FSQ,ISELEC,DM,QMIN,QMAX

VARIABLE	TYPE	PURPOSE
NCONTOR	integer	the number of contour lines to be drawn
FSP,FSQ	real	free stream pressure and mach number
ISELEC	integer	specifies velocity vectors or contour type
DM	real	specifies spacing between contours
QMIN,QMAX	real	minimum and maximum quantities to be contoured

8. COMMON /SCL3D/ XTRAD,XRAD,XOR(3)

VARIABLE	TYPE	PURPOSE
XTRAD	real	number of points in the graphics area
XRAD	real	viewing volume radius
XOR(3)	real	offsetting values for the terminal coordinates

9. COMMON/SEGGR/IREAD,NSEG,IRS,INF,INL,IXTRP

VARIABLE	TYPE	PURPOSE
IREAD	integer	unit number that EPS reads it's commands from
NSEG	integer	last assigned segment number
IRS	integer	specifies writing inputs to a run stream
INF,INL	integer	first and last points in specified color map
IXTRP	integer	specifies extrapolation of contours

FUNCTION DESCRIPTIONS

1. FUNCTION DP(IA,A,IB,B)

VARIABLE	TYPE	PURPOSE
IA	integer	starting position in array A
A(9)	real	dummy argument (3 by 3 matrix)
IB	integer	starting position in array B
B(9)	real	dummy argument (3 by 3 matrix)

This function performs a dot product on a specified element of two matrices.

2. FUNCTION ICOLD(D)
   COMMON/SEGGR/IREAD,NSEG,IRS,INF,INL,IXTRP
   COMMON/RANGE/FRANGE(300),NRANGE,ICOORD(4,256)

VARIABLE	TYPE	PURPOSE
D	real	contains the magnitude of the contour line or velocity vector

This function determines the color of the velocity vector or contour line according to its magnitude.

3. FUNCTION IX3(X,I)
   COMMON /SCL3D/ XTRAD,XRAD,XOR(3),CRMAT(9)

VARIABLE	TYPE	PURPOSE
X	real	real x, y, or z coordinate
I	integer	specifies x, y, or z coordinate

This function converts the coordinates of each grid point into terminal coordinates prior to plotting.

SUBROUTINE DESCRIPTIONS

1. SUBROUTINE CAPS(A)

VARIABLE	TYPE	PURPOSE
A(5)	character	dummy string containing EPS command

This subroutine converts all EPS commands to upper case.

2. SUBROUTINE CONTR(X,Y,Z,XC,YC,ZC,NJ,NI,K,XA,XB,YA,YB,VARS, IMAX,JMAX,KMAX,TA3,ICP)

```
COMMON/DSTN/N4,N5,N6,N7,N8,N9
COMMON/SEGGR/IREAD,NSEG,IRS,INF,INL,IXTRP
COMMON/RXNS/NCONTOR,FSP,FSQ,ISELEC,DM,QMIN,QMAX
COMMON/RANGE/ FRANGE(300),NRANGE,ICOORD(4,256)
COMMON/PRINT/iotpt,ISLU2
```

VARIABLE	TYPE	PURPOSE
X,Y,Z	real	grid for current block
XC,YC,ZC	real	cell centered coordinates
NI,NJ	integer	constant plane size
K	integer	constant plane number
XA,XB,YA,YB	integer	constant plane dimensions
VARS	real	constant plane flow quantities
IMAX,JMAX,KMAX	integer	grid dimensions for current block
TA3	real	flow quantities for current block
ICP	integer	specifies constant plane type

This subroutine plots the contours by first converting the three dimensional arrays into a set of two dimensional arrays that contain only the specified constant plane. This is done immediately for the flow quantity; however, the grid points are transformed into a cell centered system as they are placed in the new arrays. An extrapolation is then performed on the cell centered system when specified. The next step is to search the constant plane looking for the flow quantities that are on the contour lines and to draw these to the screen by calling subroutine LINE.

3. SUBROUTINE DEFMAC(IKEY,LEN,IASC)

VARIABLE	TYPE	PURPOSE
IKEY	integer	number of key to be programmed
LEN	integer	length of following array
IASC(21)	integer	ASCII code

This subroutine programs a specified key with a command specified in its ASCII equivalent.

4. SUBROUTINE DIMN(IMAX,JMAX,KMAX,MEMAX,LAP,IFTYP)
COMMON/DMNS/I1X,I1Y,I1Z,I1A,I1B,I1C,ID1,ID2,ID3,ID4,IUSED
COMMON/DSTN/N4,N5,N6,N7,N8,N9

VARIABLE	TYPE	PURPOSE
IMAX,JMAX,KMAX	integer	grid dimensions for the current block
MEMAX	integer	length of the ribbon vector
LAP(2,3)	integer	flow field dimensioning parameters
IFTYP	integer	file type

This subroutine calculates the storage locations used in the ribbon vector.

5. SUBROUTINE GTSGDM(ISI,ISEG,IRP,IDM,IST)
common / ginger/lsig,ltsig,lgfc,lgrf

VARIABLE	TYPE	PURPOSE
ISI	integer	leading signature character
ISEG	integer	segment number
IRP	integer	ASCII equivalent of report type
IDM	integer	segment dimensionality code
IST	integer	terminating signature character

This subroutine catches the report segment dimensionality report that is sent by the terminal.

6. SUBROUTINE GTV3 (SIGCHR,KEYCHR,JX,JY,JZ,JSEG,JPIK,JVIEW)
COMMON/GLOBAL/ICORD,IABSXY
COMMON/GINGER/LSIG,LTSIG,LGFC,LGRF

VARIABLE	TYPE	PURPOSE
SIGCHR	character	signature character
KEYCHR	character	keys pressed to cause report
JX,JY,JZ	integer	terminal coordinates
JSEG	integer	picked segment number
JPIK	integer	pick ID of segment
JVIEW	integer	graphics view number

This subroutine is used to catch a three dimensional graphics input (GIN) report sent by the terminal.

7. SUBROUTINE FINDMAX(X,Y,Z,TA1,TA2,TA3,TA4,IMAX,JMAX,KMAX,
IB,BMAX)

```
COMMON/DSTN/N4,N5,N6,N7,N8,N9
COMMON/SEGGR/IREAD,NSEG,IRS,INF,INL,IXTRP
COMMON/RXNS/NCONTOR,FSP,FSQ,ISELEC,DM,QMIN,QMAX
COMMON/RANGE/ FRANGE(300),NRANGE,ICOORD(4,256)
```

VARIABLE	TYPE	PURPOSE
X,Y,Z	real	grid coordinates for current block
TA1,TA2,TA3,TA4	real	flow quantities for current block
IMAX,JMAX,KMAX	integer	dimensions of the current block
IB	integer	the current block number
BMAX	integer	number of blocks on current file

The purpose of this subroutine is to find the minimum and maximum magnitudes of the flow quantity of interest. Once this is done the spacing between contours is determined and subroutine SCALE is called to draw the scale.

8. SUBROUTINE INTL(IDEL)

```
COMMON/SEGGR/IREAD,NSEG,IRS,INF,INL,IXTRP
COMMON /SCL3D/ XTRAD,XRAD,XOR(3),CRMAT(9)
COMMON/RANGE/ FRANGE(300),NRANGE,ICOORD(4,256)
```

VARIABLE	TYPE	PURPOSE
IDEL	integer	indicates the *CLS command

This subroutine initializes the terminal graphics and draws the cursors and the primary velocity vector used by EPS.

9. SUBROUTINE IORY(IS,IN,ICO)

VARIABLE	TYPE	PURPOSE
IS	integer	segment number
IN	integer	length of the following array
ICO	integer	ASCII equivalent of report code

This subroutine causes the terminal to send a segment dimensionality report for a specified segment.

10. SUBROUTINE KEYS

This subroutine contains the ASCII codes to program the function keys for shading.

11. SUBROUTINE LINE(X,Y,Z)

VARIABLE	TYPE	PURPOSE
X(2),Y(2),Z(2)	real	specify the starting and ending point of a line segment

This subroutine draws the contour lines one segment at a time.

12. SUBROUTINE PBOD(X,Y,Z,A,B,C,IXYZ,I4NORY,NI,NJ,K,XA,XB,
YA,YB,IMAX,JMAX,KMAX,ICLR,ICP,IDIR)
COMMON/PRINT/iotpt,ISLU2

VARIABLE	TYPE	PURPOSE
X,Y,Z	real	grid for current block
A,B,C	real	constant plane arrays
IXYZ	integer	terminal coordinates
I4NORY	integer	normals to the plane
NI,NJ	integer	constant plane size
K	integer	constant plane number
XA,XB,YA,YB	integer	constant plane dimensions
IMAX,JMAX,KMAX	integer	grid dimensions for current block
ICLR	integer	constant plane color
ICP	integer	specifies constant plane type
IDIR	integer	specifies the normal direction

This subroutine places the constant plane on a set of two dimensional arrays and calculates the normals to the constant plane. The coordinates are then transformed into terminal coordinates and constant plane is plotted.

13. SUBROUTINE P3D(X,Y,Z,A,B,C,IAA,IMAX,JMAX,KMAX,
TA1,TA2,TA3,TA4,CHOSE,IBL)
COMMON/DSTN/N4,N5,N6,N7,N8,N9
COMMON/SEGGR/IREAD,NSEG,IRS,INF,INL,IXTRP
COMMON/RXNS/NCONTOR,FSP,FSQ,ISELEC,DM,QMIN,QMAX
COMMON /SCL3D/ XTRAD,XRAD,XOR(3),CRMAT(9)
COMMON/RANGE/ FRANGE(300),NRANGE,ICOORD(4,256)

VARIABLE	TYPE	PURPOSE
X,Y,Z	real	grid for current block
A,B,C	real	constant plane arrays
IAA	integer	array for normals and terminal coordinates
IMAX,JMAX,KMAX	integer	grid dimensions for current block
TA1,TA2,TA3,TA4	real	flow quantities for current block
CHOSE	character	contains command issued
IBL	integer	current block number

This subroutine controls the plotting routines for grid and restart files. The dimensions of the constant plane are specified here and the appropriate subroutine is called to do the plotting. When a grid is to be plotted the constant plane is broken into ten by ten sections to increase the size of the constant plane that the terminal will accept.

14. SUBROUTINE RDGRID(X,Y,Z,IMAX,JMAX,KMAX,IP)

VARIABLE	TYPE	PURPOSE
X,Y,Z	real	grid for current block
IMAX,JMAX,KMAX	integer	grid dimensions for current block
IP	integer	current block number

The purpose of this subroutine is to read both the grid and the surface files.

15. SUBROUTINE RDRES(X,Y,Z,TA1,TA2,TA3,TA4,IMAX,JMAX,
KMAX,IBL)
COMMON/RXNS/NCONTOR,FSP,FSO,ISELEC,DM,QMIN,QMAX
COMMON/DSTN/N4,N5,N6,N7,N8,N9

VARIABLE	TYPE	PURPOSE
X,Y,Z	real	grid for current block
TA1,TA2,TA3,TA4	real	flow quantities for current block
IMAX,JMAX,KMAX	integer	grid dimensions for current block
IBL	integer	current block number

This subroutine reads the restart file and performs the calculations necessary to convert the flow field quantities into the selected contour type or velocity vectors.

16. SUBROUTINE RDXYP(IX,IY,IZ)
COMMON/GLOBAL/ICORD,IABSXY
COMMON/GINGER/LSIG,LTSIG,LGFC,LGRF

VARIABLE	TYPE	PURPOSE
IX,IY,IZ	integer	terminal coordinates

This subroutine catches the xy+ reports sent by the terminal.

17. SUBROUTINE SCALE
COMMON/SEGGR/IREAD,NSEG,IRS,INF,INL,IXTRP
COMMON/RANGE/ FRANGE(300),NRANGE,ICOORD(4,256)

This subroutine draws the scale for the contours and the velocity vectors.

18. SUBROUTINE SURF(AA,IAA,MEMAX)
COMMON/SEGGR/IREAD,NSEG,IRS,INF,INL,IXTRP
COMMON /SCL3D/ XTRAD,XRAD,XOR(3),CRMAT(9)

VARIABLE	TYPE	PURPOSE
AA	real	the ribbon vector
IAA	integer	array for normals and terminal coordinates
MEMAX	integer	length of the ribbon vector

This is the controlling subroutine for the surface plotting.

19. SUBROUTINE TEXT(FKEEP,ISTS,ISTF,ISTC)
COMMON/SEGGR/IREAD,NSEG,IRS,INF,INL,IXTRP

VARIABLE	TYPE	PURPOSE
FKEEP	integer	contains previously loaded fonts
ISTS(3)	integer	specifies text size
ISTF	integer	current text font
ISTC	integer	current text color

This subroutine reads the text then loads the correct font, sets the text color, and scales the text. Subroutine WHEELS is then called to position the text.

20. SUBROUTINE TNS3(JS,JX,JY,JZ)
COMMON /GLOBAL/ICORD,IABSXY

VARIABLE	TYPE	PURPOSE
JS	integer	segment number
JX,JY,JZ	integer	terminal coordinates

This subroutine is used to set the positions of three dimensional segments.

21. SUBROUTINE WHEELS(IT,JS,JX,JY,IOP,ICRS,KEY)
COMMON/GINGER/LSIG,LTSIG,LGFC,LGRF

VARIABLE	TYPE	PURPOSE
IT	integer	specifies graphic rubberbanding
JS	integer	segment number
JX,JY	integer	terminal coordinates
IOP	integer	graphics operation code
ICRS	integer	graphics cursor segment number
KEY	integer	key pressed to cause report

This subroutine is used to position two dimensional segments with the thumb wheels.

22. SUBROUTINE WHEEL3(IT,JS,JX,JY,JZ,IOP,ICRS)
    COMMON /SCL3D/ XTRAD,XRAD,XOR(3),CRMAT(9)
    COMMON/GINGER/LSIG,LTSIG,LGFC,LGRF

VARIABLE	TYPE	PURPOSE
IT	integer	specifies graphic rubberbanding
JS	integer	segment number
JX,JY,JZ	integer	terminal coordinates
IOP	integer	graphics operation code
ICRS	integer	graphics cursor segment number
KEY	integer	key pressed to cause report

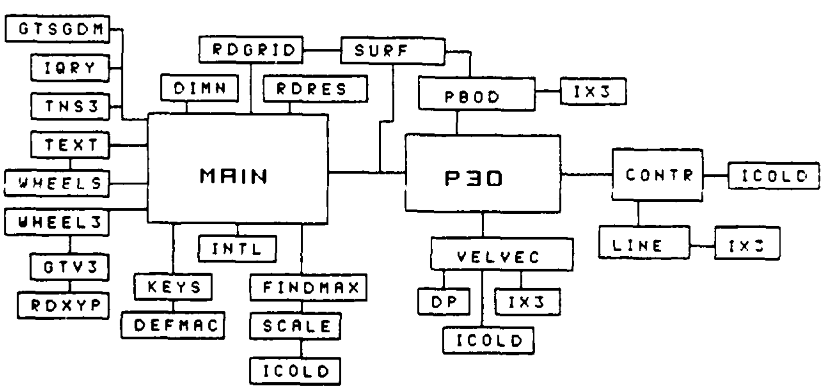
This subroutine is used to position three dimensional segments with the thumb wheels.

23. SUBROUTINE VELVEC(X,Y,Z,V,U,VEL,W,IMAX,JMAX,KMAX,
                     IA,IB,JA,JB,KK,ICP)
    COMMON/GLOBAL/ICOR,IABSXY
    COMMON/DSTN/N4,N5,N6,N7,N8,N9
    COMMON /SCL3D/ XTRAD,XRAD,XOR(3),CRMAT(9)
    COMMON/SEGGR/IREAD,NSEG,IRS,INF,INL,IXTRP
    COMMON/RXNS/NCONTOR,FSP,FSQ,ISELEC,DM,QMIN,QMAX
    COMMON/RANGE/ FRANGE(300),NRANGE,ICOORD(4,256)
    COMMON/PRINT/iotpt,ISLU2

VARIABLE	TYPE	PURPOSE
X,Y,Z	real	grid for current block
U,V,W,VEL	real	velocities for the current block
IMAX,JMAX,KMAX	integer	grid dimensions for current block
IA,IB,JA,JB	integer	constant plane dimensions
KK	integer	constant plane number
ICP	integer	specifies constant plane type

This subroutine determines the orientation of the velocity vector, its color according to magnitude, and places the vectors at the cell centers.

The following diagram shows the subroutine/function interaction.



REFERENCES

1. Lijewski, L., Cipolla, J., and Thompson, J., "Program EAGLE User's Manual, Volume I: Introduction & Grid Applications. United States Air Force Armament Laboratory. AFATL-TR-88-117.

2. Mounts, J., Belk, D., and Whitfield, D., "Program EAGLE User's Manual, Volume IV: Multi - Block, Implicit, Steady - State Euler Code". United States Air Force Armament Laboratory. AFATL-TR-88-117.

3. 4110/4120 SERIES COMMAND REFERENCE WITH 3D. Tektronix, Copywrite 1985.

4. 4100 SERIES CP/M-86 SYSTEM OVERVIEW AND DTI. Tektronix, Copywrite 1985.

5. 4115B/M4115B COMPUTER DISPLAY TERMINAL. Tektronix, Copywrite 1984.

6. 4128/4129 COLOR GRAPHICS WORKSTATION. Tektronix, Copywrite 1985.